

**METHOD AND APPARATUS FOR HIGH FREQUENCY DIGITAL CARRIER
SYNTHESIS FROM PLURAL INTERMEDIATE CARRIER WAVEFORMS**

BACKGROUND OF THE INVENTION

1. Technical Field

The present invention pertains to signal synthesizers. In particular, the present invention pertains to a digital synthesizer that generates a high frequency signal from a plurality of intermediate frequency waveforms.

2. Discussion of Related Art

Digital communication transmitters operate by modulating a carrier waveform with digital data. One possible way to generate carrier waveforms in digital systems is by utilizing a direct digital synthesizer (DDS). The DDS may be implemented by a field programmable gate array (FPGA), an application specific integrated circuit (ASIC) or other logic device and includes a phase accumulator, a phase-to-amplitude converter and a digital-to-analog converter (DAC). The phase accumulator is typically an accumulator that increments a phase value by a phase increment each time interval. The phase increment is determined by the sampling frequency and carrier frequency. The phase-to-amplitude converter is generally implemented by a sine look-up table read-only memory (ROM) that receives phase information from the phase accumulator and produces digital amplitude values. The DAC converts the digital amplitude values to an analog waveform. A conventional DDS generates a sine wave or carrier waveform. Phase, frequency and/or amplitude modulation may be performed in the digital domain within the DDS and/or within the analog domain subsequent to the digital to analog conversion.

The Nyquist rate defines the minimum sampling rate needed to generate a digital carrier waveform as twice the frequency of that waveform. Generally, realistic applications require a sampling rate greater than twice the carrier frequency to produce a viable signal. With the high frequencies required for most digital communications applications (e.g., approximately 800 MHz for cellular applications, approximately 1500 MHz for a Global Positioning System (GPS) L1

1 type signal, etc.), it is impractical and inefficient to employ an ASIC, FPGA or other logic device
2 operating at a sampling frequency of more than twice the frequency of the carrier waveform.

3 **OBJECTS AND SUMMARY OF THE INVENTION**

4 Accordingly, it is an object of the present invention to generate a high frequency signal
5 from a plurality of intermediate frequency waveforms.

6 It is another object of the present invention to employ a plurality of digital synthesizers in
7 parallel to generate intermediate frequency waveforms and to selectively combine the
8 intermediate waveforms to produce a resulting high frequency signal.

9 Yet another object of the present invention is to generate a modulated high frequency
10 signal from a plurality of modulated intermediate frequency signals.

11 The aforesaid objects may be achieved individually and/or in combination, and it is not
12 intended that the present invention be construed as requiring two or more of the objects to be
13 combined unless expressly required by the claims attached hereto.

14 According to the present invention, a signal synthesizer produces a high speed or high
15 frequency carrier waveform without employing an ASIC, FPGA or other logic device operating at
16 a sampling rate of greater than twice the carrier frequency. The signal synthesizer basically
17 simulates a high speed or high frequency direct digital synthesizer with a plurality of low speed or
18 low frequency direct digital synthesizers. The low speed synthesizers are operated in parallel and
19 each one produces an intermediate carrier waveform with a frequency less than the desired carrier
20 frequency. The intermediate carrier waveforms are subsequently multiplexed together to form a
21 high frequency digital carrier waveform that is subsequently converted to an analog signal with a
22 high-speed digital-to-analog converter. In addition, the signal synthesizer may perform phase,
23 frequency and/or amplitude modulation.

24 The above and still further objects, features and advantages of the present invention will
25 become apparent upon consideration of the following detailed description of specific
26 embodiments thereof, particularly when taken in conjunction with the accompanying drawings
27 wherein like reference numerals in the various figures are utilized to designate like components.

28 **BRIEF DESCRIPTION OF THE DRAWINGS**

1 Fig. 1 is a schematic block diagram of the general architecture of a signal synthesizer
2 according to the present invention.

3 Fig. 2 is a graphical illustration of a plurality of intermediate frequency waveforms and the
4 manner in which these waveforms are combined to produce a high frequency waveform in
5 accordance with the present invention.

6 Fig. 3 is a schematic block diagram of an exemplary signal synthesizer of the present
7 invention implemented by a Field Programmable Gate Array (FPGA).

8 **DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS**

9 A signal synthesizer according to the present invention is illustrated in Fig. 1. Initially, the
10 signal synthesizer produces a high frequency carrier signal or other waveform without operating a
11 device (e.g., ASIC, FPGA, logic device, etc.) at a sampling rate of greater than twice the
12 frequency of that carrier signal or other waveform. The signal synthesizer basically simulates a
13 high-speed or high frequency direct digital synthesizer (DDS) by employing a plurality of low-
14 speed or low frequency direct digital synthesizers. Specifically, signal synthesizer 10 includes a
15 plurality of digital synthesizer modules 20, a multiplexer 16 and a digital-to-analog converter 18.
16 Digital synthesizer modules 20 are arranged in parallel with each synthesizer module producing a
17 digital waveform with an intermediate frequency less than the frequency of the resulting carrier
18 signal or other waveform as described below. The output of each digital synthesizer module is
19 coupled to multiplexer 16. The multiplexer combines the intermediate frequency waveforms into
20 a digital signal corresponding to the desired carrier signal or other waveform as described below.
21 The multiplexer is coupled to digital-to-analog converter 18 that converts the resulting digital
22 signal to the desired analog carrier signal or waveform. The signal synthesizer may be
23 implemented by various devices (e.g., FPGA, ASIC, logic devices, etc.) and/or circuitry with
24 conventional or custom components.

25 The quantity of intermediate waveforms required by synthesizer 10 to generate the carrier
26 waveform depends upon the maximum sampling rate of the device (e.g., ASIC, FPGA, logic
27 device, etc.) implementing the synthesizer. In particular, the minimum Nyquist or sampling rate,
28 f_s , for a desired carrier or other waveform having a frequency, f_c , may be expressed as follows:
29

$$f_s > 2 * f_c. \quad (\text{Equation 1})$$

The quantity of intermediate waveforms, k , utilized to produce the desired waveform for a device (e.g., ASIC, FPGA, logic device, etc.) with a maximum sampling rate, f_{\max} , may be expressed as follows:

$$k = \lceil f_s / f_{\max} \rceil, \quad (\text{Equation 2})$$

where k is rounded up to the nearest integer and f_s is the minimum sampling rate as described above. Since each digital synthesizer module 20 typically produces an intermediate waveform, k digital synthesizer modules are typically employed by synthesizer 10 to produce k intermediate waveforms in order to generate the desired carrier signal or other waveform.

Each digital synthesizer module 20 includes a phase accumulator 12 and a phase-to-amplitude converter 14. The phase accumulator is preferably implemented by an accumulator that increments a phase value by a phase increment each time interval or clock corresponding to an intermediate sampling frequency or rate. This enables the phase accumulator to incrementally cycle through the intermediate waveform. The phase increment is derived from the intermediate sampling rate and the frequency of the desired carrier signal or other waveform as described below. The intermediate sampling rate, f_{is} , may be expressed as follows:

$$f_{is} = f_s / k, \quad (\text{Equation 3})$$

where f_s is the minimum sampling rate as described above and k is the quantity of intermediate waveforms utilized to produce the desired carrier signal or other waveform as described above. The frequency of the intermediate signal or waveform, f_{ic} , generated by each synthesizer module 20 may be expressed as:

$$f_{ic} = \langle f_c \rangle_{f_{is}}, \quad (\text{Equation 4})$$

where f_c is the frequency of the desired carrier signal or other waveform as described above, f_{is} is the intermediate sampling rate as described above and $\langle f_c \rangle_{f_{is}}$ represents the modulus operation which computes the remainder of f_c/f_{is} . The modulus operation basically enables an intermediate waveform to oscillate and encompass desired carrier signal or other waveform points when the intermediate waveform is not an integer multiple of or aligned with the desired waveform. In other words, the modulus operation compensates for any frequency offsets between the intermediate waveform and an integer multiple of the desired waveform.

A corresponding phase offset for the particular digital synthesizer module is applied to the accumulated phase value within the phase accumulator to produce a waveform phase value for conversion to a corresponding waveform amplitude value by phase-to-amplitude converter 14. The amplitude value basically corresponds to the waveform amplitude for the particular phase of the waveform. The phase offsets enable generation of a set of intermediate waveforms that are successively shifted in phase by the phase offset and collectively include the points or samples of the desired carrier signal or other waveform as described below. The desired waveform points are subsequently selected from the appropriate intermediate waveforms by multiplexer 16 to generate the desired carrier signal or other waveform as described below. The corresponding phase offset applied by a digital synthesizer module phase accumulator is basically a multiple of the phase offset, $\Delta\phi_{\text{offset}}$, which may be expressed as:

$$\Delta\phi_{\text{offset}} = (f_c/k)/f_{is}, \quad (\text{Equation 5})$$

where f_c is the frequency of the desired carrier signal or other waveform as described above, k is the quantity of intermediate waveforms utilized to produce the desired carrier signal or other waveform as described above and f_{is} is the intermediate sampling rate as described above.

Phase accumulator 12 within each digital synthesizer module 20 produces the waveform phase value at each time interval or clock corresponding to the intermediate sampling frequency. The waveform phase value, $\Delta\phi_j(n)$, produced by each digital synthesizer module 20 for a corresponding intermediate waveform, j , may be expressed as:

$$\Delta\phi_j(n) = 2*\pi*f_{ic}*n + 2*\pi*\Delta\phi_{offset}*j, \text{ (Equation 6)}$$

where j is an integer from 0 to $k - 1$, k is the quantity of intermediate waveforms utilized to produce the desired carrier signal or other waveform as described above, f_{ic} is the intermediate waveform frequency as described above, n is the sample number or time and $\Delta\phi_{offset}$ is the phase offset as described above. The term $\Delta\phi_{offset}*j$ basically represents the successive phase shift of the intermediate waveforms or multiple of the phase offset applied by phase accumulator 12 of synthesizer modules 20. The waveform values, $\Delta\phi_j(n)$, produced by synthesizer modules 20 are each applied to a corresponding phase-to-amplitude converter 14 to ascertain a corresponding waveform amplitude value.

Phase-to-amplitude converter 14 is typically in the form of a memory and receives the intermediate waveform values produced by a corresponding phase accumulator 12. Converter 14 is preferably implemented as a sine/cosine lookup table read only memory (ROM) and stores sine and/or cosine values. The converter basically receives the intermediate waveform information from the corresponding phase accumulator and provides the appropriate sine and/or cosine value for each intermediate waveform sample, where the converter value represents the intermediate waveform amplitude for that sample. The output of converter 14, $Carrier_j(n)$, for each digital synthesizer module 20 to produce a corresponding intermediate waveform, j , may be expressed as:

$$Carrier_j(n) = \cos(\Delta\phi_j(n)) = \cos(2*\pi*f_{ic}*n + 2*\pi*\Delta\phi_{offset}*j); \text{ or} \quad \text{(Equation 7)}$$

$$Carrier_j(n) = \sin(\Delta\phi_j(n)) = \sin(2*\pi*f_{ic}*n + 2*\pi*\Delta\phi_{offset}*j), \quad \text{(Equation 8)}$$

where j is an integer from 0 to $k - 1$ as described above, k is the quantity of intermediate waveforms utilized to produce the desired carrier signal or other waveform as described above, $\Delta\phi_j(n)$ is the waveform value as described above, f_{ic} is the intermediate carrier or waveform frequency as described above, n is the sample number or time as described above and $\Delta\phi_{offset}$ is the phase offset as described above. The intermediate waveform is preferably generated as a

1 cosine wave; however, converters 14 may provide sine and/or cosine values in order to generate
2 that waveform in any fashion (e.g., cosine wave, sine wave, combination of sine and cosine
3 waves, etc.).

4 Phase and/or frequency modulation may further be performed by synthesizer modules 20.
5 For example, phase accumulator 12 of each synthesizer module 20 may further include a phase
6 modulation module 11. This module basically applies a modulation phase offset value,
7 $\text{phaseoffset}(n)$, to the phase offset that represents a modulated phase at a corresponding sample.
8 The output of converter 14 in the case of phase modulation, $\text{PhaseModCarrier}_j(n)$, for each digital
9 synthesizer module 20 to produce a corresponding modulated intermediate waveform, j , may be
10 expressed as:

11
12
$$\text{PhaseModcarrier}_j(n) = \cos(2*\pi*f_{ic}*n + 2*\pi*\Delta\phi_0*j + \text{phaseoffset}(n)); \text{ or } \text{ (Equation 9)}$$

13
14
$$\text{PhaseModcarrier}_j(n) = \sin(2*\pi*f_{ic}*n + 2*\pi*\Delta\phi_0*j + \text{phaseoffset}(n)), \text{ (Equation 10)}$$

15
16 where j is an integer from 0 to $k - 1$ as described above, k is the quantity of intermediate
17 waveforms utilized to produce the desired signal or waveform as described above, f_{ic} is the
18 intermediate waveform frequency as described above, n is the sample number or time as
19 described above, $\Delta\phi_{\text{offset}}$ is the phase offset as described above and $\text{phaseoffset}(n)$ is the
20 modulation phase offset value as described above. The modulated intermediate waveform is
21 preferably generated as a cosine wave; however, converters 14 may provide sine and/or cosine
22 values in order to generate the modulated waveform in any fashion (e.g., cosine wave, sine wave,
23 combination of sine and cosine waves, etc.).

24 Moreover, phase accumulator 12 of each synthesizer module 20 may further include a
25 frequency modulation module 15. This module basically applies a frequency offset, $\text{freqoffset}(n)$,
26 that represents a modulated frequency at a corresponding sample. The output of converter 14 in
27 the case of frequency modulation, $\text{FreqModCarrier}_j(n)$, for each digital synthesizer module 20 to
28 produce a corresponding modulated intermediate waveform, j , may be expressed as:

$$\text{FreqModCarrier}_j(n) = \cos (2*\pi*(f_{ic} + \text{freqoffset}(n))*n + 2*\pi*\Delta\phi_{\text{offset}} *j); \text{ or (Equation 11)}$$

2

$$\text{FreqModCarrier}_j(n) = \sin (2*\pi*(f_{ic} + \text{freqoffset}(n))*n + 2*\pi*\Delta\phi_{\text{offset}} *j), \quad (\text{Equation 12})$$

4

5 where j is an integer from 0 to k – 1 as described above, k is the quantity of intermediate
6 waveforms utilized to produce the desired signal or waveform as described above, f_{ic} is the
7 intermediate waveform frequency as described above, n is the sample number or time as
8 described above, $\Delta\phi_{\text{offset}}$ is the phase offset as described above and $\text{freqoffset}(n)$ is the modulated
9 frequency offset value as described above. The modulated intermediate waveform is preferably
10 generated as a cosine wave; however, converters 14 may provide sine and/or cosine values in
11 order to generate the modulated waveform in any fashion (e.g., cosine wave, sine wave,
12 combination of sine and cosine waves, etc.).

13 In addition, the digital synthesizer modules may apply amplitude modulation to the digital
14 waveforms produced by converters 14. Specifically, the synthesizer modules may include an
15 amplitude modulation module 17 to produce amplitude modulated values for selection by
16 multiplexer 16. In this case, converters 14 each provide amplitude values to amplitude
17 modulation module 17. The amplitude modulation module applies a constant and/or function,
18 k_{Amp} , to the amplitude values to produce a resulting modulated amplitude value. The modulated
19 amplitude value, $\text{AmpModCarrier}_j(n)$, produced by each digital synthesizer module 20 for a
20 corresponding modulated intermediate waveform, j, may be expressed as:

21

$$\text{AmpModCarrier}_j(n) = k_{\text{Amp}} * \cos (\Delta\phi_j(n)); \text{ or} \quad (\text{Equation 13})$$

23

$$\text{AmpModCarrier}_j(n) = k_{\text{Amp}} * \sin (\Delta\phi_j(n)), \quad (\text{Equation 14})$$

25

26 where j is an integer from 0 to k – 1 as described above, k is the quantity of intermediate
27 waveforms utilized to produce the desired signal or waveform as described above, k_{Amp} is a
28 constant or function for the amplitude modulation, n is the sample number or time as described
29 above and $\Delta\phi_j(n)$ is the waveform phase value as described above. Phase, frequency and

1 amplitude modulation may be applied to an intermediate waveform by a synthesizer module 20
2 either individually or in any combination.

3 The digital waveforms from digital synthesizer modules 20 are selectively combined by
4 multiplexer 16 to produce a digital waveform corresponding to the desired carrier signal or other
5 waveform as illustrated, by way of example only, in Fig. 2. Specifically, the intermediate
6 waveform samples are generated by synthesizer modules 20 (Fig. 1) at each time interval or clock
7 corresponding to the intermediate sampling frequency. An initial intermediate waveform is
8 generated without a phase offset, while the phases of the remaining intermediate waveforms are
9 successively offset from the initial waveform by a multiple of the phase offset, $\Delta\phi_{\text{offset}}$, as
10 described above (e.g., Equations 6, 7 and 8). The phase offsets basically shift the intermediate
11 waveforms successively from the initial waveform to enable the intermediate waveforms to
12 encompass points or samples of the desired carrier signal or other waveform. The phase offset is
13 determined to evenly space the intermediate waveforms within an intermediate sampling interval
14 (e.g., the phase offset basically includes the appropriate portion of the desired carrier frequency
15 attributed to an intermediate waveform (f_c/k) divided by the intermediate sampling frequency).
16 Since k minimum sampling intervals, $1/f_s$, are within an intermediate sampling interval, $1/f_{is}$ (e.g.,
17 $1/f_{is} = k*1/f_s$ derived from Equation 3), each successive intermediate waveform provides a sample
18 of the desired waveform at each successive minimum sampling interval within an intermediate
19 sampling interval. Thus, the phase offsets enable the intermediate waveforms to provide a desired
20 waveform sample at each time interval or clock corresponding to the minimum sampling rate.

21 For example, the graph of Fig. 2 includes a desired waveform 30 and five intermediate
22 waveforms 40, 41, 42, 43 and 44. By way of example only, the carrier frequency, f_c , is 33 MHz
23 with a minimum sampling rate, f_s , of 100 MHz, maximum device sampling rate, f_{max} , of 20 MHz,
24 an intermediate sampling rate, f_{is} , of 20 MHz, an intermediate waveform frequency, f_{ic} , of 13
25 MHz and the quantity of intermediate waveforms, k , of five. The intermediate waveforms are
26 illustrated as inverted 7MHz (-7MHz) sine waveforms. In effect, these signals yield the
27 intermediate waveform frequency, f_{ic} , of 13MHz since these signals are applied with respect to
28 the 20MHz sampling signal, f_{is} , of the device. The points or samples for the intermediate
29 waveforms are generated at each intermediate sampling time interval or clock by digital

1 synthesizer modules 20 as described above. Intermediate waveform 40 is generated without a
2 phase offset, while intermediate waveforms 41, 42, 43, 44 are generated with a successive
3 multiple of the phase offset, $\Delta\phi_{\text{offset}}$, as described above. The phase offsets basically shift initial
4 waveform 40 successively to produce values for samples of the desired waveform at each
5 successive minimum sampling time interval or clock within the intermediate sampling interval. In
6 other words, each intermediate waveform in succession provides a desired waveform sample at a
7 successive minimum sampling time interval or clock. Thus, the intermediate waveforms
8 collectively provide samples of the desired waveform at the minimum sampling rate within each
9 intermediate sampling interval.

10 Multiplexer 16 selects appropriate intermediate waveform samples to generate the desired
11 waveform. Basically, multiplexer 16 selects a sample from each intermediate waveform
12 successively within each intermediate sampling interval. Since the minimum sampling interval is
13 k times greater than the intermediate sampling interval, the multiplexer selects an intermediate
14 waveform sample from a successive intermediate waveform at each minimum sampling time
15 interval or clock. Appropriate samples from the intermediate waveforms may be selected for
16 subsequent intermediate sampling intervals in substantially the same manner described above,
17 thereby retrieving the desired waveform samples from the intermediate waveforms in a cyclical
18 fashion. Multiplexer 16 is driven by a counter (not shown) to cycle through and select samples
19 from the intermediate waveforms (or synthesizers). The digital waveform sample selected by
20 multiplexer 16 is applied to digital-to-analog converter 18 to produce the corresponding portion
21 of the desired analog carrier signal or other waveform at each minimum sampling interval.

22 An exemplary embodiment of synthesizer 10 implemented in the form of a field
23 programmable gate array (FPGA) is illustrated in Fig. 3. Specifically, synthesizer 10 is in the
24 form of an FPGA including a plurality of digital synthesizer modules 20, a central or common
25 phase accumulator 50 to provide a phase value to each synthesizer module 20 and a series of
26 multiplexers 68. The synthesizer further includes a multiplexer 16 and digital-to-analog converter
27 18 on a separate circuit board and coupled to the FPGA. The phase accumulator includes an
28 adder 52 and a register 54 to store the resulting accumulated phase value from adder 52. The
29 adder receives the previously accumulated phase value from register 54 and a phase increment

1 from an increment register 53 and produces a current accumulated phase value that is stored in
2 register 54 at each time interval or clock corresponding to the intermediate sampling rate. The
3 resulting accumulated phase value from register 54 is applied to a truncate module 55 to remove
4 extraneous result bits. By way of example only, the truncate module receives a thirty-two bit
5 phase value and produces a fourteen bit result by removing the eighteen least significant bits
6 (LSBs). The truncated value is applied to an adder 57 that receives phase modulation information
7 from a phase modulation multiplexer 56. Multiplexer 56 is controlled by a register (not shown) to
8 select either a zero or a phase modulation value for entry into adder 57. The phase modulation
9 value is generated by logic (not shown) and provided at an appropriate data rate. Multiplexer 56
10 basically serves as a switch to enable or disable phase modulation. A setting to provide a zero
11 value into adder 57 basically enables the adder to provide the accumulated phase value unchanged
12 for distribution to the digital synthesizer modules, thereby effectively disabling phase modulation.
13 Conversely, a setting to provide a phase modulation value enables application of that value to the
14 accumulated phase value and enablement of phase modulation (e.g., Equations 9 and 10). Adder
15 57 and register 56 basically form phase modulation module 11 of synthesizer modules 20 (Fig. 1)
16 described above. The resulting phase value from adder 57 is applied to an adder 22 of each
17 digital synthesizer module 20.

18 Adder 22 of each synthesizer module is further coupled to an offset register 59 that
19 contains the phase offset for that synthesizer module (e.g., $\Delta\phi_{\text{offset}} * j$). Since the phase offset for
20 the initial waveform or synthesizer module is zero (e.g., $\Delta\phi_{\text{offset}} * j = 0$ for the initial synthesizer
21 module, where j for that synthesizer module equals zero), an offset register and adder is not
22 needed for that synthesizer module. Adder 22 combines the phase value and phase offset to
23 produce the waveform value, $\Delta\phi_j(n)$, for phase-to-amplitude converter 14 as described above
24 (e.g., Equation 6). Alternatively, a single offset register may be employed to determine the phase
25 offset values for each synthesizer module by using a single offset value and a series of
26 multiplication logic.

27 Converter 14 of each digital synthesizer module 20 includes an output for cosine values
28 and an output for sine values. By way of example only, the outputs include twelve bits. Each
29 converter output is applied to a corresponding multiplier 62, 64 that receives amplitude

1 modulation information from an amplitude modulation multiplexer 58. Multiplexer 58 is in the
2 form of two multiplexers, each associated with a corresponding multiplier 62, 64. The
3 multiplexers are each controlled by a register (not shown) to select either a value of one or an
4 amplitude modulation value for entry into multipliers 62, 64. The amplitude modulation values
5 are generated by logic (not shown) and provided at an appropriate data rate. With respect to
6 amplitude modulation, the cosine values produced by converter 14 correspond to the In-Phase (I)
7 components, while the sine values correspond to the Quadrature (Q) components. The amplitude
8 modulation values typically include I and Q data streams that are applied to the corresponding I
9 and Q components. Multiplexer 58 basically serves as a switch to enable or disable amplitude
10 modulation. A setting to provide a one value to the multipliers disables amplitude modulation
11 since the converter values are multiplied by one and effectively passed unchanged. Conversely, a
12 setting to provide amplitude modulation values to the multipliers enables amplitude modulation
13 and application of those values to the corresponding converter outputs. The resulting products of
14 multipliers 62, 64 are applied to an adder 66 to produce waveform amplitude values for
15 multiplexer 16. Multipliers 62, 64, amplitude modulation multiplexer 58 and adder 66 basically
16 form amplitude modulation module 17 (Fig. 1) described above, where the amplitude modulated
17 values, $\text{AmpModCarrierIQ}_j(n)$, may be expressed as:

$$18 \quad \text{AmpModCarrierIQ}_j(n) = k_1 * \cos(\Delta\phi_j(n)) + k_2 * \sin(\Delta\phi_j(n)), \quad (\text{Equation 15})$$

19
20
21 where j is an integer from 0 to $k - 1$ as described above, k is the quantity of intermediate
22 waveforms utilized to produce the desired signal or waveform as described above, k_1 and k_2 are
23 values or I and Q data streams (e.g., constants and/or functions) for the amplitude modulation, n is
24 the sample number or time as described above and $\Delta\phi_j(n)$ is the waveform phase value as
25 described above.

26 Multiplexers 56 and 58 may be switched at a fast rate in order to provide both phase and
27 amplitude modulation. The multiplexers may be switched inversely (e.g., one multiplexer is
28 permitted to enable the corresponding modulation at one time or, in other words, phase
29 modulation is enabled when amplitude modulation is disabled and phase modulation is disabled

1 when amplitude modulation is enabled) to attain a constant envelope signal with both phase and
2 amplitude modulation. Frequency modulation may be performed as described above (e.g.,
3 Equations 11 and 12) by adjusting the phase increment within increment register 53 at an
4 appropriate rate to apply a frequency offset.

5 In this exemplary embodiment, the cosine and sine values are combined by adder 66 to
6 produce the waveform amplitude value for each synthesizer module. When amplitude modulation
7 is absent, the summation of the sine and cosine values produces the intermediate waveform, and
8 hence the resulting carrier waveform, with a phase shift of forty-five degrees; however, the phase
9 shift is typically irrelevant to receivers that may receive the resulting carrier signal. Alternatively,
10 converters 14 may provide cosine and/or sine values in any fashion to generate the intermediate
11 waveforms. The resulting values from each pair of adjacent synthesizer modules 20 is applied to
12 a corresponding multiplexer 68. The multiplexer alternately selects values from the synthesizer
13 modules as output from the FPGA and is typically driven by a counter (not shown). These
14 outputs are coupled to multiplexer 16 and converter 18, typically implemented by a commercially
15 available digital-to-analog converter with a corresponding 4:1 multiplexer. Multiplexer 16 is
16 typically driven by a counter (not shown). Multiplexers 68 basically enable multiplexer 16 to
17 function as an 8:1 multiplexer and are typically driven to provide the alternate values at a rate
18 twice that of the intermediate sampling rate.

19 Multiplexers 68 alternately select values as described above, where multiplexer 16 selects
20 a value from each successive multiplexer 68 each minimum sampling interval or clock.
21 Multiplexer 16 basically selects values from each multiplexer 68 in a cyclical fashion until a
22 waveform value from each synthesizer module has been retrieved during an intermediate
23 sampling interval. Thus, values from each of successive k waveforms are selected by multiplexer
24 16 during an intermediate sampling interval, where a waveform value is available and selected by
25 multiplexer 16 at each minimum sampling interval or clock within the intermediate sampling
26 interval. The order of waveform selection may be controlled by the offsets placed in offset
27 registers 59 and via the selection control lines of multiplexers 16, 68. Each sample selected by
28 multiplexer 16 is converted to an analog signal by digital-to-analog converter 18 at the minimum
29 sampling rate (f_s from Equation 1) to produce the desired analog carrier signal or other waveform

1 as described above. The minimum sampling interval or clock is preferably provided to converter
2 18, where synthesizer 10 may derive the appropriate clock signals for operation from that
3 minimum sampling clock. The FPGA may be configured to operate at any desired rate and utilize
4 any quantity of intermediate waveforms.

5 The FPGA is typically coupled to a user interface (e.g., GUI, microprocessor, etc.) to
6 enable a user to specify the desired waveform. The interface determines the appropriate
7 information (e.g., offsets, rates, increments, modulation values, etc.) as described above and
8 downloads the information to the FPGA for subsequent operation. The signal synthesizer may
9 apply phase, frequency and/or phase modulation to the intermediate waveforms either
10 individually or in any combination.

11 It will be appreciated that the embodiments described above and illustrated in the
12 drawings represent only a few of the many ways of implementing a method and apparatus for high
13 frequency digital carrier synthesis from plural intermediate carrier waveforms.

14 The synthesizer may include any quantity of any types of synthesizer modules generating
15 intermediate waveforms at any desired frequencies. The intermediate waveforms may be selected
16 or produced to partition the desired carrier signal or other waveform in any desired manner (e.g.,
17 an even or uniform carrier frequency distribution among intermediate waveforms, a non-uniform
18 carrier frequency distribution among intermediate waveforms, any quantity of intermediate
19 waveforms of any frequency, etc.). The synthesizer may be implemented by any types of
20 conventional or other devices (e.g., ASIC, FPGA, logic devices, etc.) or circuitry (e.g., processors,
21 chips, etc.). The synthesizer may utilize any quantity of intermediate waveforms to generate the
22 desired carrier signal or other waveform, where each synthesizer module may produce any
23 quantity of intermediate waveforms. The various variables or indices mentioned herein (e.g., 'j',
24 'k', etc.) are preferably integer values and may contain integers within any desired range;
25 however, these variables may be of any type and contain any desired values.

26 The synthesizer components (e.g., phase accumulator, phase-to-amplitude converter,
27 analog-to-digital converter, modulation modules, etc.) may be of any quantity and may be
28 implemented by any conventional or other components or circuitry performing the functions
29 described herein. Alternatively, the synthesizer may be implemented by any quantity of

1 processors (e.g., microprocessor, etc.) to determine amplitude values for and/or generate any
2 quantity of intermediate waveforms. The processors may further include a digital-to-analog
3 converter to produce the desired carrier signal or waveform. The phase accumulator may be
4 implemented by any quantity of any conventional or other devices accumulating a value (e.g.,
5 ALU, adder, etc.). Alternatively, the phase accumulator may include components (e.g., adders,
6 ALU, multipliers, dividers, shifters, logic circuitry, etc.) to produce a phase value based on any
7 mathematical and/or logical operations on input values (e.g., multiplication, division, shifts,
8 subtraction, etc.). The increment and value registers may be of any quantity and may be
9 implemented by any type of conventional or other storage or memory device (e.g., register, buffer,
10 memory, etc.) and store any desired values. The phase accumulator may apply any type of phase
11 or other offsets (e.g., phase modulation, frequency modulation, phase shift for intermediate
12 waveforms, etc.) to the accumulated phase value. The output of the phase accumulator may
13 include any quantity of bits for any desired resolution. The synthesizer may include a common
14 phase accumulator or any quantity of phase accumulators to provide phase values for the digital
15 synthesizers.

16 The phase-to-amplitude converter may be implemented by any type of memory or storage
17 device (e.g., ROM, RAM, buffers, CORDIC algorithm, etc.) and may contain any desired
18 information (e.g., sine values, cosine values, any mathematical information, etc.). Alternatively,
19 the converter may be implemented by any quantity of devices or processors (e.g., microprocessor,
20 circuitry, etc.) to produce the desired values. The phase-to-amplitude converter may include any
21 quantity of outputs each providing any types of values (e.g., sine values, cosine values or any
22 other mathematical values or information). The output of the phase-to-amplitude converter may
23 indicate the amplitude value in any fashion (e.g., sine value, cosine value, other mathematical
24 value or any combination thereof). The phase-to-amplitude converter may store amplitude values
25 with various modulation (e.g., frequency, phase, amplitude, etc.) incorporated into those values to
26 provide a modulated waveform value. The synthesizer may include a common phase-to-amplitude
27 converter or any quantity of phase-to-amplitude converters to provide amplitude values for the
28 waveform. The phase-to-amplitude converter may accommodate any quantity of phase values

1 and provide corresponding amplitude values. The phase-to-amplitude converter may provide
2 values including any quantity of bits for any desired resolution.

3 The digital synthesizer modules may perform any type of modulation (e.g., phase,
4 frequency, amplitude, etc.) and/or encoding to produce modulated and/or encoded signals. The
5 phase, frequency and amplitude modulation may be performed individually or in any desired
6 combination and may utilize any offsets, constants or functions. The modulation and/or encoding
7 may be performed in the digital and/or analog domain or in any combination thereof. The
8 modulated and/or encoded values may include any quantity of bits for any desired resolution.

9 The synthesizer multiplexer may be of any quantity and may be implemented by any
10 conventional or other multiplexers or selection devices (e.g., circuitry, logic, buffers, registers,
11 etc.). The multiplexer may accommodate any quantity of inputs or outputs and include any
12 quantity of control lines. The control lines may be driven by any suitable devices (e.g., counters,
13 circuitry, processor, etc.). The multiplexer may select output from the digital synthesizer modules
14 in any desired order or fashion suitable to produce the desired carrier signal or waveform.

15 The digital-to-analog converter may be of any quantity and may be implemented by any
16 conventional or other digital-to-analog converter (e.g., chip, circuitry, resistive ladder, etc.). The
17 converter may process the amplitude samples at any desired rate.

18 The FPGA synthesizer may be implemented by any type of FPGA, ASIC, logic device or
19 other circuitry. The FPGA may include the DAC and/or multiplexer and operate at any desired
20 rate (e.g., generally less than the sampling rate of the desired carrier signal or waveform). The
21 FPGA components (e.g., adders, registers, truncate module, multiplier, waveform selection and
22 modulation multiplexers, ROM, clock dividers, etc.) may be implemented by any conventional or
23 other components performing the functions described herein. The various values (e.g., phase,
24 modulation, truncation, etc.) may include any quantity of bits for any desired resolution. The
25 waveform selection and/or DAC multiplexers may be combined or integrated in any fashion (e.g.,
26 a single 8:1 multiplexer may be employed, etc.), while the waveform selection, modulation and
27 DAC mutliplexers may accommodate selection from any quantity of inputs. The multiplexers
28 may be driven by any conventional or other devices (e.g., counters, logic, processor, etc.). The
29 FPGA may include any quantity of registers to store any quantity of any types of values (e.g.,

1 phase offsets, modulation, etc.). The FPGA and/or synthesizer may include any types of modules
2 (e.g., logic, circuitry, etc.) and/or be coupled to devices (e.g., interfaces, logic, processors, etc.) to
3 provide and/or determine any information (e.g., offsets, rates, increments, modulation values,
4 etc.) for generating the desired carrier signal or waveform. The FPGA and/or synthesizer may
5 include any types of modules (e.g., clocks, etc.) and/or be coupled to devices to produce
6 appropriate clock signals for operation. The phase, frequency and/or amplitude modulation may
7 be enabled or disabled in any fashion (e.g., switches, multiplexers, mathematically, logically,
8 etc.).

9 The present invention is not limited to the applications disclosed herein, but may be
10 utilized in any high speed communications or other system to generate a carrier signal or other
11 waveform requiring a sampling rate greater than a device (e.g., FPGA, ASIC, logic device, etc.)
12 operational rate. The present invention effectively multiplies the device sampling rate by the
13 quantity of parallel synthesizer modules employed (e.g., and is generally limited by the logic
14 space of the device and sampling rate of the digital-to-analog converter).

15 From the foregoing description, it will be appreciated that the invention makes available a
16 method and apparatus for high frequency digital carrier synthesis from plural intermediate carrier
17 waveforms, wherein a digital synthesizer generates a high frequency signal from a plurality of
18 intermediate frequency waveforms.

19 Having described preferred embodiments of a new and improved method and apparatus
20 for high frequency digital carrier synthesis from plural intermediate carrier waveforms, it is
21 believed that other modifications, variations and changes will be suggested to those skilled in the
22 art in view of the teachings set forth herein. It is therefore to be understood that all such
23 variations, modifications and changes are believed to fall within the scope of the present
24 invention as defined by the appended claims.